

7. Übungsblatt (erschienen am 25.01.2017)

Aufgabe 7.1 (Votieraufgabe)

Sei $\Omega \subset \mathbb{R}^n$ ein beschränktes Gebiet mit Lipschitz-Rand, $T > 0$ und $f \in L^2((0, T) \times \Omega)$. Wir betrachten die parabolische partielle Differentialgleichung

$$u_t(x, t) - \Delta u(x, t) + u(x, t) = f(x, t) \quad \forall x \in \Omega, t \in (0, T) \quad (1)$$

mit Anfangswerten $u(x, 0) = u_0(x) \in L^2(\Omega)$ und homogenen Dirichletrandwerten. Zeigen Sie, dass genau ein $u \in L^2(0, T, H_0^1(\Omega))$ existiert, das (1) löst.

Hinweis: Verwenden Sie die variationelle Formulierung aus Bemerkung 4.3(b) (iv), um dann mit dem Satz von Lions-Lax-Milgram für $H := L^2(0, T, H_0^1(\Omega))$ und $V := \{v \in W(0, T, H_0^1(\Omega), H^{-1}(\Omega)) : v(\cdot, T) = 0\}$, wobei die V -Norm durch $\|v\|_V^2 := \|v\|_{L^2(0, T, H_0^1(\Omega))}^2 + \|v(0)\|_{L^2(\Omega)}^2$ definiert ist, die Existenz zu zeigen. Dabei kann die partielle Integrationsformel aus Lemma 4.2 hilfreich sein.

Aufgabe 7.2 (Programmieraufgabe zum Votieren)

Wir übernehmen das Gebiet Ω und das Teilgebiet D des 6. Übungsblattes und betrachten die parabolische partielle Differentialgleichung

$$u_t - \Delta u = \chi_D, \quad u|_{\partial\Omega} = 0, \quad u(x, 0) = 0. \quad (2)$$

Wir wollen die *vertikale Linienmethode* verwenden, um eine Approximation an die wahre Lösung zu erhalten. Dazu wird zuerst bezüglich des Ortes diskretisiert. Mit $b(u, v) := \int_{\Omega} \nabla u \cdot \nabla v \, dx$ lautet die variationelle Formulierung von (2) (vgl. Bemerkung 4.3 (b) (iii)): finde $u \in W(0, T, H_0^1(\Omega), H^{-1}(\Omega))$, sodass $u(x, 0) = 0$ und

$$\langle \dot{u}(t), v \rangle + b(u(t), v) = \int_{\Omega} \chi_D(x)v(x) \, dx \quad \forall v \in H_0^1(\Omega), t \in (0, T) \text{ fast überall.}$$

Wir wählen wieder $V^T \subset H_0^1(\Omega)$ (den Raum der stetigen und stückweise linearen Funktionen, vgl. Übungsblatt 5) als Ansatzraum und suchen eine Funktion $u_N : [0, T] \rightarrow V^T$, mit $u_N(0) = 0$ und

$$\int_{\Omega} u_N(t)v \, dx + b(u(t), v) = \int_{\Omega} \chi_D(x)v(x) \, dx \quad \forall v \in V^T, t \in (0, T) \text{ fast überall.}$$

Offenbar gilt $u_N(t) = \sum_{j=1}^m \eta_j(t)\Lambda_j$, mit den *Koeffizientenfunktionen* $\eta_j : [0, T] \rightarrow \mathbb{R}$ und $u_N(t)$ löst genau dann das obige variationelle Problem, wenn die Koeffizientenfunktionen das Differentialgleichungssystem

$$\sum_{j=1}^m \dot{\eta}_j(t) \int_{\Omega} \Lambda_j \Lambda_k \, dx + \sum_{j=1}^m \eta_j(t) b(\Lambda_j, \Lambda_k) = \int_{\Omega} \chi_D(x) \Lambda_k(x) \, dx \quad \forall k = 1, \dots, m$$

lösen. Mit der Notation

$$y(t) := (\eta_j(t))_{j=1}^m : [0, T] \rightarrow \mathbb{R}^m \quad h(t) := \left(\int_{\Omega} \chi_D(x) \Lambda_k(x) dx \right)_{k=1}^m : [0, T] \rightarrow \mathbb{R}^m$$
$$G := \left(\int_{\Omega} \Lambda_j \Lambda_k dx \right)_{j,k=1}^m \in \mathbb{R}^{m \times m} \quad B := (b(\Lambda_j, \Lambda_k))_{j,k=1}^m \in \mathbb{R}^{m \times m}$$

gilt es also die gewöhnliche Differentialgleichung

$$\dot{y}(t) = G^{-1}(h(t) - By(t)) \quad (3)$$

zu lösen.

Schreiben Sie eine MATLAB-Funktion, welche zu einer Zeitschrittweite $\tau > 0$ eine Approximation an die wahre Lösung von (2) bestimmt, indem eine Approximation an (3) bestimmt wird mit:

- (a) dem expliziten Euler-Verfahren, d.h. $\frac{y(t+\tau) - y(t)}{\tau} \approx G^{-1}(h(t) - By(t))$.
- (b) dem impliziten Euler-Verfahren, d.h. $\frac{y(t+\tau) - y(t)}{\tau} \approx G^{-1}(h(t + \tau) - By(t + \tau))$.
- (c) dem Crank-Nicholson-Verfahren, d.h. $\frac{y(t+\tau) - y(t)}{\tau} \approx \frac{1}{2} G^{-1}((h(t) - By(t)) + (h(t + \tau) - By(t + \tau)))$.

Verwenden Sie jeweils die Zeitschrittweite $\tau = \frac{1}{10}$ und berechnen Sie die Approximationen der ersten 1000 Zeitschritte. Visualisieren und vergleichen Sie ihre Ergebnisse.

Hinweise zur Übungsblattbearbeitung:

- Zu **Votieraufgaben** wird keine schriftliche Abgabe verlangt. Die Lösung wird in der Übung besprochen.
- Alle Aufgaben von Übungsblatt 7 werden in der Übung am 06.02.2017 besprochen. **Die Übung am 09.02.2017 entfällt!**
- Eine **Programmieraufgabe zum Votieren** soll wie eine normale Votieraufgabe sinnvoll bearbeitet/gelöst werden. Diese wird wie eine normale Votieraufgabe in den Übungen besprochen und diskutiert. Bringen Sie dazu ihre Ausarbeitungen möglichst auf einem USB-Stick mit in die Übung oder schicken Sie diese dem Übungsleiter.